

Analiza obiektowa

3@KASK

16 czerwca 2009

Symbol projektu: 3@KASK	Opiekun projektu: mgr inż. Tomasz Boiński
Nazwa Projektu: Wizualizacja grafów za pomocą biblioteki Prefuse	

Nazwa Dokumentu: Analiza obiektowa	Nr wersji: 0.0
Odpowiedzialny za dokument: Piotr Kunowski	Data pierwszego sporządzenia: 23 maja 2009
Przeznaczenie: DLA KLIENTA	Data ostatniej aktualizacji: 16 czerwca 2009

Historia dokumentu

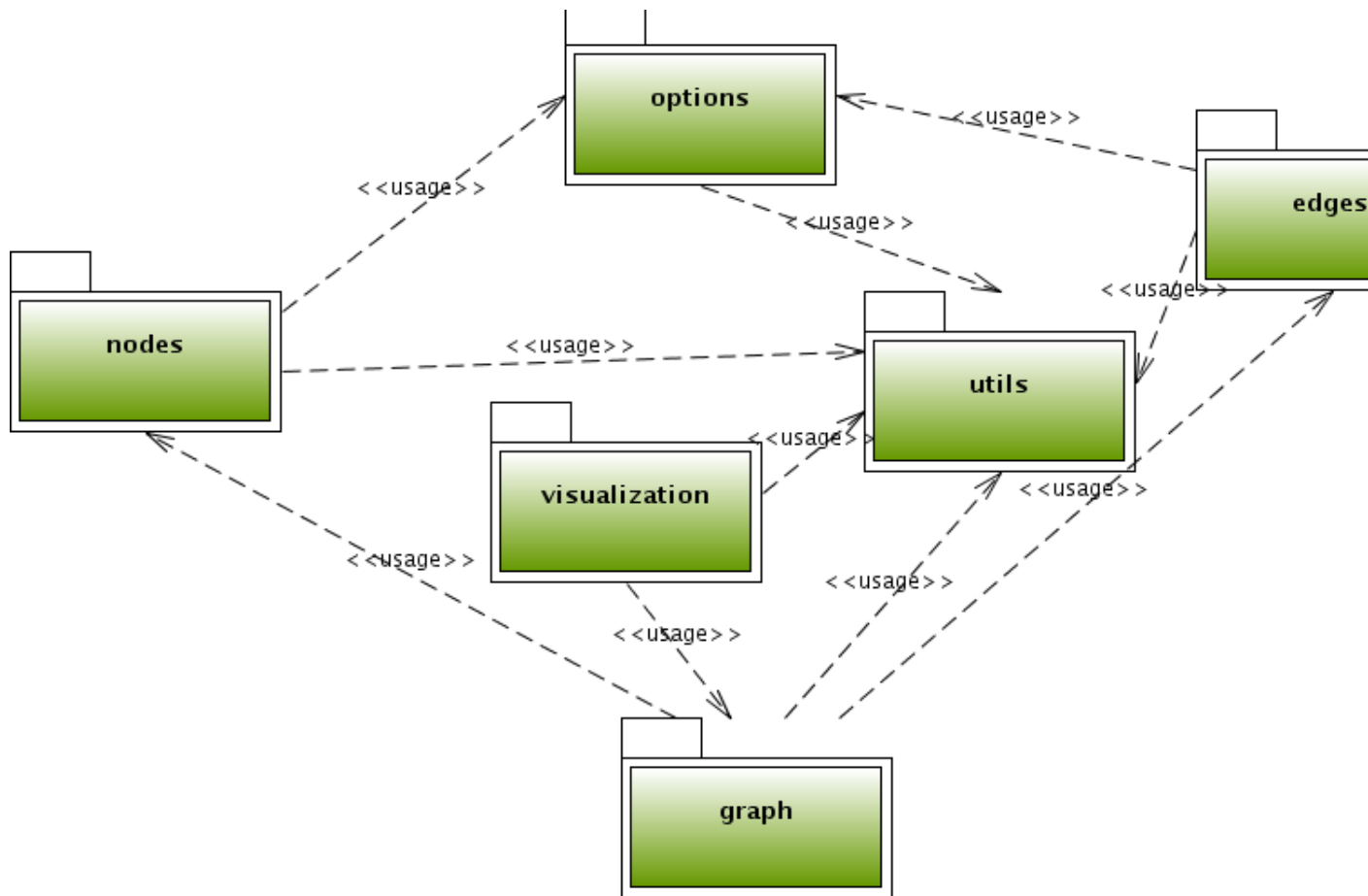
Wersja	Opis modyfikacji	Rozdział/strona	Autor modyfikacji	Data
1	Stworzenie	wszystkie	Grupa projektowa	23.05.09

Spis treści

1	Pakiety	3
1.1	Diagram	3
1.2	Opis pakietów	3
2	Pakiet options	5
2.1	Diagram	5
2.2	Opis klasy	5
3	Pakiet nodes	7
3.1	Diagram	7
3.2	Opis klasy	7
4	Pakiet edges	12
4.1	Diagram	12
4.2	Opis klasy	12
5	Pakiet visualization	15
5.1	Diagram	15
5.2	Opis klasy	15
6	Pakiet graph	16
6.1	Diagram	16
6.2	Opis klasy	17
7	Pakiet utils	17
7.1	Diagram	17
7.2	Opis klasy	17

1 Pakiety

1.1 Diagram



1.2 Opis pakietów

P001	options
Opis:	Pakiet zawierający klasy z polami opisującymi różne (modyfikowalne) ustawienia wizualizacji takie jak: kolory, grubość linii itp.
Interfejsy:	
Realizowane wymagania:	WF002, WF001, WI004
Priorytet:	średnio ważne

P002	nodes
Opis:	Pakiet z klasami odpowiedzialnymi za wizualizację i przechowywanie danych o wierzchołkach.
Interfejsy:	
Realizowane wymagania:	WF004, WF005, WF006, WF007, WI004
Priorytet:	bardzo ważne

P003	edges
Opis:	Pakiet z klasami odpowiedzialnymi za wizualizację i przechowywanie danych o krawędziach.

Interfejsy:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	bardzo ważne

P004	visualization
Opis:	Zawiera dodatkowe klasy przydatne w wizualizacji.
Interfejsy:	
Realizowane wymagania:	WF001, WF008, WI004
Priorytet:	średnio ważne

P005	graph
Opis:	Pakiet zawiera klasy, które zawierają podstawowe operacje na danych OwlApi oraz graph.
Interfejsy:	
Realizowane wymagania:	WD001
Priorytet:	bardzo ważne

P006	utils
Opis:	Pakiet zawiera klasy pomocnicze
Interfejsy:	
Realizowane wymagania:	CF005
Priorytet:	bardzo ważne

2 Pakiet options

2.1 Diagram



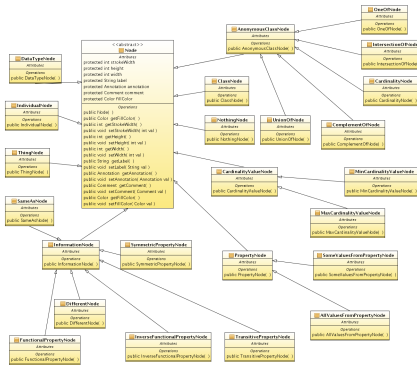
2.2 Opis klasy

CO001	EdgeColors
Opis:	Zawiera definicje kolorów dla poszczególnych rodzajów krawędzi.
Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none"> • domainEdgeColor • edgeColor • equivalentEdgeColor • equivalentPropertyEdgeColor • functionalEdgeColor • inverseOfEdgeColor • propertyEdgeColor • rangeEdgeColor • subEdgeColor
Metody:	
Realizowane wymagania:	WF002
Priorytet:	średnio ważny
CO002	NodeColors

Opis:	Zawiera definicje kolorów dla poszczególnych rodzajów krawędzi.
Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none"> • allValuesFromNodeColor • cardinalityNodeColor • cardinalityValueNodeColor • classNodeColor • complementOfNodeColor • dataTypeNodeColor • differentNodeColor • functionalPropertyNodeColor • individualNodeColor • informationNodeColor • intersectionOfNodeColor • inverseFunctionalNodeColor • maxCardinalityValueNodeColor • minCardinalityValueNodeColor • nothingNodeColor • oneOfNodeColor • propertyNodeColor • sameAsNodeColor • someValuesFromNodeColor • symmetricPropertyNodeColor • thingNodeColor • transitivePropertyNodeColor • unionOfNodeColor
Metody:	
Realizowane wymagania:	WF002
Priorytet:	średnio ważny

3 Pakiet nodes

3.1 Diagram



3.2 Opis klasy

CN001	Node
Opis:	Klasa nadrzędna względem wszystkich używanych klas obsługi wierzchołków. Zawiera definicje podstawowych pól o funkcji.
Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none"> • strokeWidth • height • width • annotation • comment • Color fillColor • String label
Metody:	
Realizowane wymagania:	WF004, WF005, WF006, WF007, WI004
Priorytet:	bardzo ważne

CN002	AllValuesFromPropertyNode
Opis:	
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WF006, WF007, WI004
Priorytet:	ważne

CN003	AnonymousClassNode
Opis:	Klasa reprezentuje wierzchołek klas anonimowych OWL.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	

Realizowane wymagania:	WF005, WI004
Priorytet:	ważne

CN004	CardinalityNode
Opis:	Klasa reprezentuje wierzchołek klas anonimowych OWL będących wynikiem ograniczenia kardynalności.
Klasy nadrzędne:	AnonymousNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF007, WI004
Priorytet:	ważne

CN005	CardinalityValueNode
Opis:	Klasa reprezentuje wierzchołek z dokładnym ograniczeniem kardynalności (OWL Cardinality).
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF007, WI004
Priorytet:	ważne

CN006	ClassNode
Opis:	Klasa reprezentuje wierzchołek OWL Class.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WF005, WI004
Priorytet:	ważne

CN007	ComplementOfNode
Opis:	Klasa reprezentuje wierzchołek klas anonimowych OWL będących wynikiem dopełnienia (OWL ComplementOf).
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CN008	DatatypeNode
Opis:	Klasa reprezentuje wierzchołek OWL DataType.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WI04
Priorytet:	ważne

CN009	DifferentNode
-------	---------------

Opis:	Klasa reprezentuje wierzchołek oznaczający relację DifferentFrom lub AllDifferent pomiędzy wystąpieniami klas (OWL Individual).
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CN010	FunctionalPropertyNode
Opis:	Klasa reprezentuje wierzchołek oznaczający, że dane OWL Property to FunctionalProperty.
Klasy nadrzędne:	InformationNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CN011	IndividualNode
Opis:	Klasa reprezentuje wierzchołek instancji OWL Individual.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WI004
Priorytet:	ważne

CN012	InformationNode
Opis:	Klasa ta jest klasą nadrzędną, dla klas wierzchołków reprezentujących informacje o różnych właściwościach OWL Property.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF010, WI004
Priorytet:	ważne

CN013	IntersectionOfNode
Opis:	Klasa reprezentuje wierzchołek klas anonimowych OWL będących wynikiem przecięcia (OWL IntersectionOf).
Klasy nadrzędne:	AnonymousNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF005, WI004
Priorytet:	ważne

CN014	inverseFunciotnalPropertyNode
Opis:	Klasa reprezentuje wierzchołek oznaczający, że dane OWL Property to InverseFunctionalProperty.
Klasy nadrzędne:	InformationNode

Atrybuty:	
Metody:	
Realizowane wymagania:	WF007, WI004
Priorytet:	ważne

CN015	MaxCardinalityValueNode
Opis:	Klasa reprezentuje wierzchołek ograniczenia OWL MaxCardinality.
Klasy nadrzędne:	CardinalityValueNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF007, WI004
Priorytet:	ważne

CN016	MinCardinalityValueNode
Opis:	Klasa reprezentuje wierzchołek ograniczenia OWL MinCardinality.
Klasy nadrzędne:	CardinalityValueNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF007, WI004
Priorytet:	ważne

CN017	NothingNode
Opis:	Klasa reprezentuje wierzchołek OWL Nothing.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WF005, WI004
Priorytet:	ważne

CN018	OneOfNode
Opis:	Klasa reprezentuje wierzchołek klas anonimowych OWL reprezentujących 1 z klas określonego zbioru (wynik OWL OneOf).
Klasy nadrzędne:	AnonymousClassNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF005, WF006, WI004
Priorytet:	ważne

CN019	PropertyNode
Opis:	Klasa reprezentuje wierzchołek OWL Property.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WF007, WI004
Priorytet:	ważne

CN020	SameAsNode
Opis:	Klasa reprezentuje wierzchołek oznaczający relację OWL SameAs pomiędzy wystąpieniami klas (OWL Individual).
Klasy nadrzędne:	InformationNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF005, WF006, WI004
Priorytet:	ważne

CN021	SomeValuesFromPropertyNode
Opis:	
Klasy nadrzędne:	PropertyNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF005, WF006, WI004
Priorytet:	ważne

CN022	SymmetricPropertNode
Opis:	Klasa reprezentuje wierzchołek oznaczający, że dane OWL Property to SymmetricProperty.
Klasy nadrzędne:	InformationNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF007, WI004
Priorytet:	ważne

CN023	ThingNode
Opis:	Klasa reprezentuje wierzchołek OWL Thing.
Klasy nadrzędne:	Node
Atrybuty:	
Metody:	
Realizowane wymagania:	WF004, WF005, WI004
Priorytet:	ważne

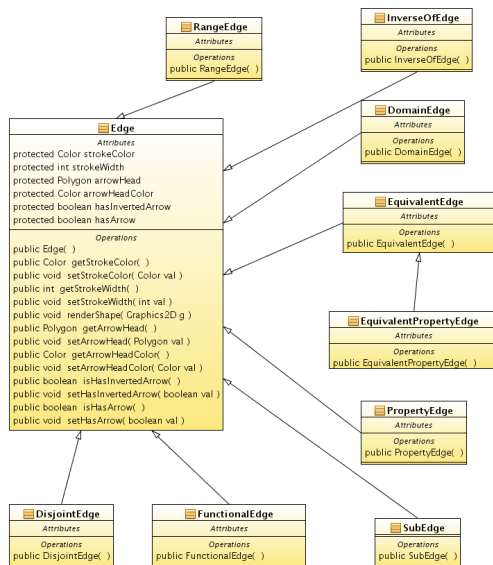
CN024	TransitivePropertyNode
Opis:	Klasa reprezentuje wierzchołek oznaczający, że dane OWL Property to TransitiveProperty.
Klasy nadrzędne:	InformationNode
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CN025	UnionOfNode
Opis:	Klasa reprezentuje wierzchołek klas anonimowych OWL będących wynikiem unii (OWL UnionOf).
Klasy nadrzędne:	AnonymousNode
Atrybuty:	

Metody:	
Realizowane wymagania:	WF005, WF006, WI004
Priorytet:	ważne

4 Pakiet edges

4.1 Diagram



4.2 Opis klasy

CE001	Edge
Opis:	Klasa reprezentująca krawędź na grafie
Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none"> • Color strokeColor • int strokeWidth • boolean hasArrow • boolean hasInvertedArrow • Polygon arrowHead • Color arrowHeadColor

Metody:	<ul style="list-style-type: none"> • <code>getStrokeColor ()</code> • <code>setStrokeColor (Color val)</code> • <code>getStrokeWidth ()</code> • <code>setStrokeWidth (int val)</code> • <code>getArrowHead()</code> • <code>setArrowHead(Polygon arrowHead)</code> • <code>isHasArrow()</code> • <code>setHasArrow(boolean hasArrow)</code> • <code>isHasInvertedArrow()</code> • <code>setHasInvertedArrow(boolean hasInvertedArrow)</code> • <code>getArrowHeadColor()</code> • <code>setArrowHeadColor(Color arrowHeadColor)</code>
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	bardzo ważne

CE002	DisjointEdge
Opis:	
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE003	DomainEdge
Opis:	
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE004	EquivalentEdge
Opis:	
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE005	EquivalentPropertyEdge
Opis:	

Klasy nadrzędne:	EquivalentEdge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE006	FunctionalEdge
Opis:	
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE007	InverseOfEdge
Opis:	
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

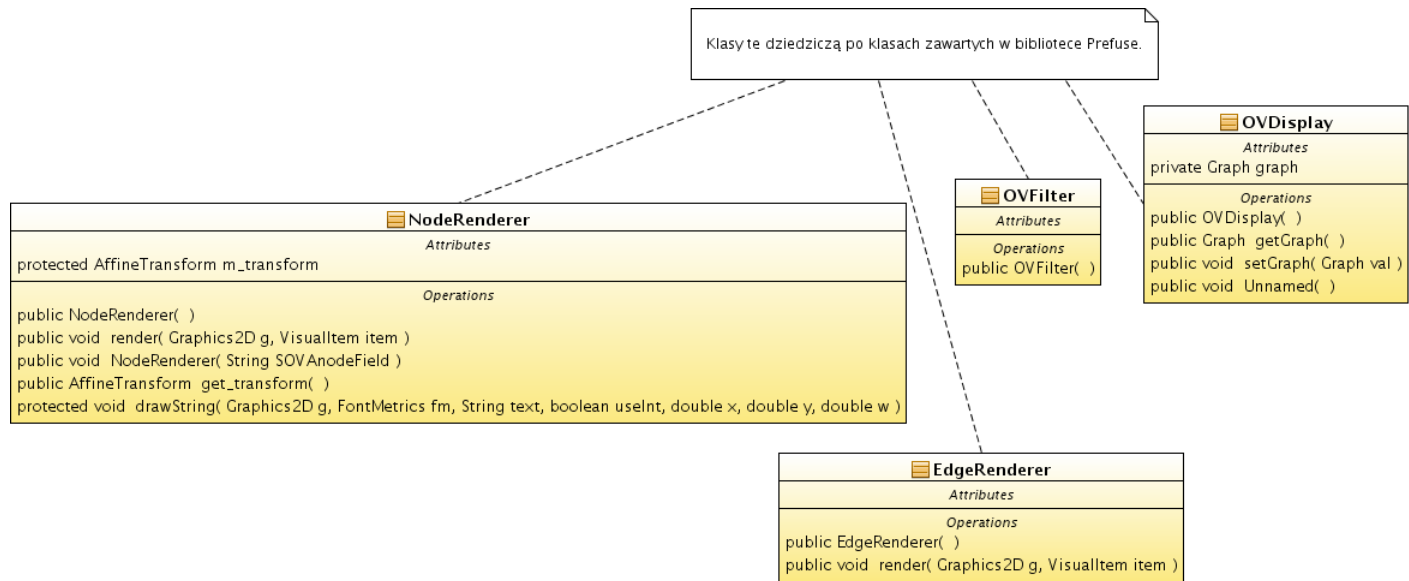
CE008	PropertyEdge
Opis:	Klasa reprezentująca na grafie relację między property a klasą
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE009	RangeEdge
Opis:	Klasa reprezentująca na grafie właściwość property Range
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

CE010	SubEdge
Opis:	Klasa reprezentująca na grafie związek SubClass pomiędzy klasami
Klasy nadrzędne:	Edge
Atrybuty:	
Metody:	
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	ważne

5 Pakiet visualization

5.1 Diagram



5.2 Opis klasy

CV001	EdgeRenderrer
Opis:	Klasa przeciążająca metody renderowania krawędzi grafu z biblioteki prefuse.
Klasy nadrzędne:	prefuse.render.EdgeRenderrer
Atrybuty:	
Metody:	<ul style="list-style-type: none"> render(Graphics2D g, VisualItem item) - metoda renderująca krawędź
Realizowane wymagania:	WF001, WF008, WI004
Priorytet:	ważne

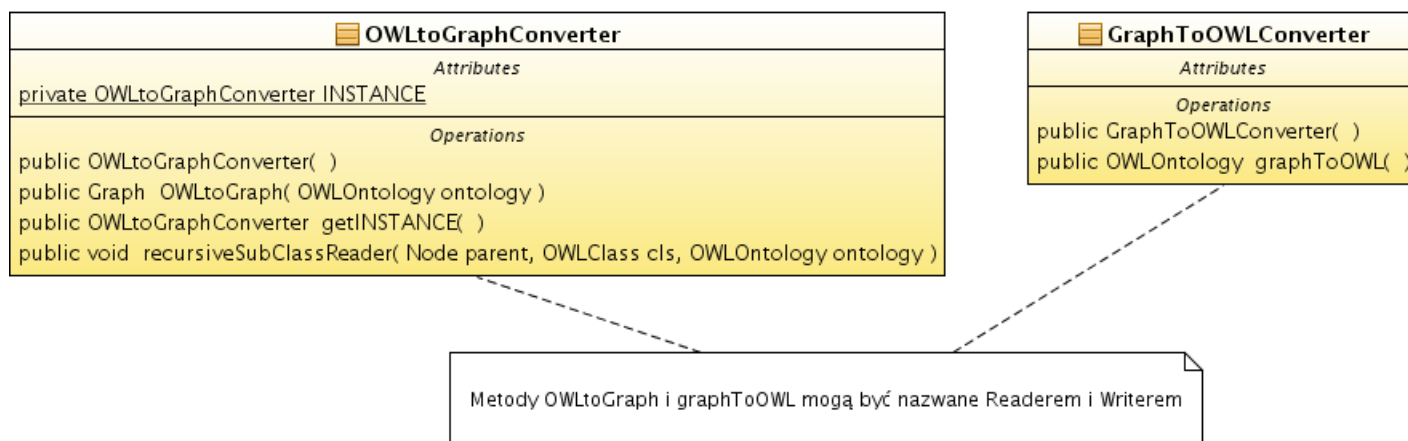
CV002	NodeRenderer
Opis:	Klasa przeciążająca metody renderowania wierzchołków grafu z biblioteki prefuse.
Klasy nadrzędne:	prefuse.render.LabelRenderer
Atrybuty:	
Metody:	<ul style="list-style-type: none"> render (Graphics2D g, VisualItem item) - metoda renderująca wierzchołek drawString(Graphics2D g, FontMetrics fm, String text, boolean useInt, double x, double y, double w) - metoda wypisująca na wierzchołku String
Realizowane wymagania:	WF001, WF008, WI004
Priorytet:	ważne

CV003	OVDisplay
Opis:	Klasa tworząca obiekt JComponent do umieszczenia na okienku JAVA zawierający wygenerowany graf z wizualizacją
Klasy nadrzędne:	prefuse.Display
Atrybuty:	<ul style="list-style-type: none"> Graph graph - obiekt typu prefuse.data.graph zawierający dane o grafie do wyświetlenia.
Metody:	<ul style="list-style-type: none"> getGraph() - zwraca graf z wyświetlanymi danymi setGraph(Graph graph) - nadpisuje obecny graf podanym generateGraphFromOWL(OWLontology ont) - wpisuje do klasy obiekt Grpah wygenerowany na podstawie ontologii
Realizowane wymagania:	WF001, WF002, WF008, WI004
Priorytet:	ważne

CV004	OVFilter
Opis:	Klasa zawierająca filtry służące do wyświetlania danych w różnych zakresach
Klasy nadrzędne:	
Atrybuty:	
Metody:	
Realizowane wymagania:	WF001, WF008, WI004
Priorytet:	ważne

6 Pakiet graph

6.1 Diagram



6.2 Opis klasy

CG001	GraphToOWLConverter
Opis:	Klasa zawierająca metody pozwalające na przetwarzanie obiektów grafów z prefuse na obiekty OWL API. Klasa jest singletonem.
Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none"> • INSTANCE - instancja klasy GraphToOWLConverter
Metody:	<ul style="list-style-type: none"> • getInstance() - zwraca instancję klasy • GraphToOWL(OWLontology ontology) -Zamienia graf z biblioteki prefuse na ontologię zapisana w OWL API.
Realizowane wymagania:	WD001, WI004
Priorytet:	ważne

CG002	OWLtoGraphConverter
Opis:	Klasa zawierająca metody pozwalające na przetwarzanie obiektów OWL API na obiekty prefuse. Klasa jest singletonem.
Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none"> • INSTANCE - instancja klasy GraphToOWLConverter
Metody:	<ul style="list-style-type: none"> • getInstance() - zwraca instancję klasy • recursiveSubClassReader(Node parent, OWLClass cls, OWLontology ontology) - wczytuje do grafu OWL wszystkie klasy wraz z ich podklasami. • OWLtoGraph(OWLontology ontology) -Zamienia ontologię w OWL API na graf z biblioteki prefuse.
Realizowane wymagania:	WD001, WI004
Priorytet:	ważne

7 Pakiet utils

7.1 Diagram

7.2 Opis klasy

CU001	Debug
Opis:	Klasa do użycia przy debugowaniu, zapewnia strumień z błędami zwracanymi przez bibliotekę. Klasa jest singletonem.

Klasy nadrzędne:	
Atrybuty:	<ul style="list-style-type: none">• INSTANCE - instancja klasy Debug• Debug - Strumień do którego wpisywane są informacje potrzebne do debugowania
Metody:	<ul style="list-style-type: none">• getInstance() - zwraca instancję klasy• setStream(PrintStream ps) - ustawia podany strumień jako strumień na który zwracane będą błędy• sendMessage(String s) - wysyła wiadomość na strumień do debugowania, jeżeli został wcześniej podpięty za pomocą funkcji setStream
Realizowane wymagania:	WF006, WF007, WI004
Priorytet:	bardzo ważne